ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Detection of Hardware Insertion Trojan in scalable Encryption Algorithm Using Path Delay

P.Vineetha
M.Tech Student
Dr.K.VSubba Reddy Institute of Technology.

S. Feroz Shah Ahmed, M.Tech
Assistant Professor,
Dr.K.VSubba Reddy Institute of Technology.

Abstract

Trusted IC design is a recently emerged topic since fabrication factories are moving worldwide in order to reduce cost. In order to get a low-cost but effective hardware Trojan detection method to complement traditional testing methods, a new behavior-oriented category method is proposedIn this paper we analyzed hardware Trojan horses insertion and detection in Scalable Encryption Algorithm (SEA) crypto. We design flow of SEA crypto and most importantly we focused on Gate level Trojan insertions. We choose path delays in order to detect Trojan at Gates level in design phase. Because the path delays detection technique is cost effective and efficient method to detect Trojan. The comparison of path delays makes small Trojan circuits significant from a delay point of view.All the synthesis and simulation results are performed using Verilog HDL.Theproposed circuit has been simulated using Xilinx ISE14.4.

*Keywords—**Hardware Trojan horses (HTH), HTH detection and insertion, Scalable Encryption Algorithm (SEA), path delay, payload Trojan.***

INTRODUCTION

With the advent of electronic commerce and portable devices for communications, cryptology has become exceedingly important science in the present day. Indeed, remote and secure data access requires the use of appropriate security methods. Modern cryptography therefore responds to this need for security but its adapted integration in the wide variety of communication systems has opened new design challenges. Cryptographic circuits are vulnerable to various side-channel attacks that target their hardware implementations to extract secret information stored inside them. Hardware Trojan Horses (HTHs or Trojan) are malicious design modifications intended to cause the design to function incorrectly.

Physical attacks which target the implementation of cryptographic circuits (in smartcards, pay-TV and SIM cards, etc.) have been known for some years now. They are widely classified as “observation” and “perturbation” attacks. Observation or side channel attacks (SCA) consist in observing physical emanations of the system, like power (Differential Power analysis, or DPA [13]) or E=H field (Electromagnetic Analysis, or EMA [14]). Thereafter statistical tools are deployed to find dependency between the predicted and observed behavior. Perturbation or fault attacks consist in the injection of faults during the execution of a cryptographic algorithm. From the knowledge of one or multiple couples (correct cipher text, faulted cipher text), some hypotheses on the secret key can be discarded. This generic attack strategy is referred to as DFA (Differential Fault Analysis). DFA is very effective against some cryptographic algorithms. For example in AES, the number of faulty cipher text required to break the key can be as low as two. There are several techniques known for fault injection in a system. The variations of the supply voltage, over clocking, temperature increase, or the irradiation by a laser beam will most probably lead to a wrong computation result that can be exploited to realize a DFA. This kind of attack represents a real threat for the implementation of cryptographic algorithms such as the AES.

In this work we propose to study the interest of hardware Trojan insertion and detection on Scalable Encryption Algorithm (SEA). SEA is a scalable encryption algorithm targeted for small embedded

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

applications. It was initially designed for software implementations in controllers, smart cards or processors. But in [1] the authors presented the importance in ASIC implementation of SEA. So, we targeted mainly on two levels to insert Trojan in the ASIC design flow of SEA crypto. One is Gate-level and another one is Layoutlevel. Compare to Gate-level the attacker has more possibility to insert Trojan at Layout-level. Due to the advancement of reverse engineering process layout of a chip can be acquired from GDSII.

In this context we have taken path delay side channel of SEA crypto in order to detect hardware Trojan at both Gatelevel and layout-level by using Fingerprint concept [6].

The rest of paper is organized as follows: Section 2 describes the SEA algorithm and its generic loop implementation. In Section 3 we discussed about VLSI design flow at abstraction level along with possible Trojan insertion levels. Based the testing procedure, the experimental setup steps are introduced in Section 4. Section 5 presents our experimental results. Finally, Conclusion and future work drawn in Section 6.

SEA ALGORITHM DESCRIPTION

SEA is a parametric block cipher for resource constrained systems (e.g. sensor networks, RFIDs) that has been introduced in [2]. It was initially designed as a low-cost encryption/authentication routine (i.e. with small code size and memory) targeted for processors with a limited instruction set (i.e. AND, OR, XOR gates, word rotation and modular addition). The algorithm takes the plaintext, key and bus sizes as parameters.

In this section we give a complete description of SEA algorithm and its loop implementation

Basic Operations and Parameters

SEAn,b operates on various text, key and word sizes. It is based on a Feistel structure with a variable number of rounds, and is defined with respect to the following parameters:

n: plaintext size, key size.
b: processor (or word) size.
nb= n/2b: number of words per Feistel branch.
nr : number of block cipher rounds.

As only constraint, it is required that n is a multiple of 6b. For example, using an 8-bit processor, we can derive a 96-bit block ciphers, denoted as SEA96,8 [2]. SEA is based on a limited number of operations denoted as follows:

(1) Bitwise XOR $\oplus$
(2) Addition mod 2^b $\boxplus$
(3) 3-bit substitution box S :={0,5,6,7,4,3,1,2}
(4) Word rotation R on n_b -word vectors:
$R: yi_{+1}=x_i, \quad 0 \le I \le n_b\text{-}2$
(5) Bit rotation r on n_b-word vectors:
$r: y_{3i} = x_{3i} >>> 1,$
$y_{3i+1} = x_{3i+1},$
$y_{3i+2} = x_{3i+2} <<< 1, \quad 0 \le I \le n_b\text{-}2$

Where >>> and <<< represents the cyclic right and left shifts inside a word.

The Round and Key Round

Based on the previous definitions, encrypt round FE, decrypt round FD and key round FK defined as:

$$[L_{i+1},R_{i+1}] = F_E(L_i,R_i,K_i) \leftrightarrow R_{i+1} = R(L_i) \oplus r(S(R_i \boxplus K_i))$$
$$L_{i+1} = R_i$$
$$[L_{i+1},R_{i+1}] = F_D(L_i,R_i,K_i) \leftrightarrow R_{i+2} = R^{-1}(L_i \oplus r(S(R_i \boxplus K_i)))$$
$$L_{i+1} = R_i$$
$$[KL_{i+1},KR_{i+1}] = F_K(KL_i,KR_i,C_i) \leftrightarrow$$
$$KR_{i+1} = KL_i \oplus R(r(S(KR_i \boxplus C_i)))$$
$$KL_{i+1} = KR_i$$

Generic Loop Architecture

Loop architecture implementation of SEA introduced in [1]. Unlike in [2] this loop architecture will support both encryption and decryption and executes one round per clock cycle. In this implementation the round function and key schedule do not share any resources. This loop architecture has benefits for FPGAs compare to [2] architecture. The structure of generic loop architecture of SEA is shown in Figure 1.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

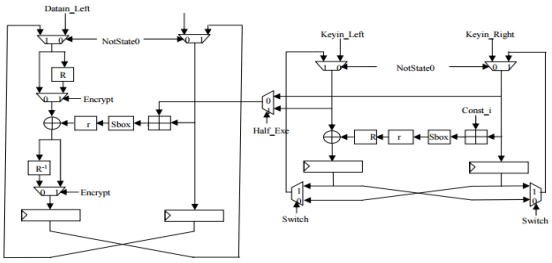


Fig.1. Generic loop architecture of SEA [1].

ABSTRACTION LEVELS-TROJAN INSERTION

The semiconductor industry has spread across borders in the time of globalization. Different design phases of an Integrated Circuit (IC) may be performed at geographically dispersed locations. This coupled with the outsourcing design and fabrication to increase profitability has become a common trend in the semiconductors industry. However, this business model comes with an ample scope of introducing malicious behavior to a part of the IC [3].

This malicious hardware is very tough to find at functional testing level. Because, the Trojan size is very less compare total chip area and it mostly never invoke for test vectors. So, we need to depend on side channels like power, delay and Electromagnetic radiation in order to detect this malicious hardware. Even though there are many Trojan detection methods are available the path delay is efficient technique for Gate-level and Layout-level Trojan insertions. We have taken the path delay to detect Trojan due to its efficiency [4].

In [8] authors classified the Trojan depend on its physical, activation and action characteristics. Trojan taxonomies that group Trojans based on their triggering and leaking mechanisms have also been developed. All of these taxonomies assume that hardware Trojans are inserted only at the fabrication phase; however, they can be inserted at other phases and have different functionalities. In [9] authors propose a new taxonomy that has a broader set of attributes. They classify Trojans according to insertion phase, abstraction level, activation mechanism, effects and location.

Here we discussed about abstraction level which gives the clear idea about possible insertion of Trojan levels in the VLSI design flow. Figure 2 shows the abstraction level of design flow along with Trojan insertion at different levels.

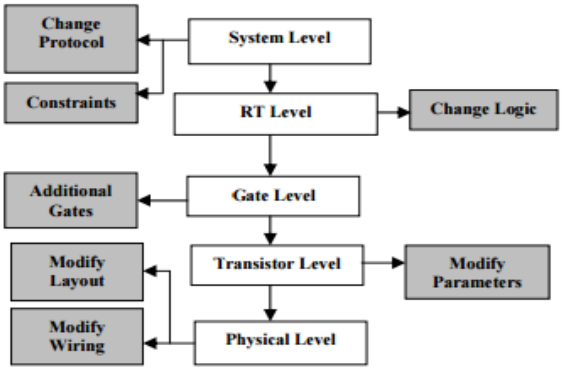


Fig.2. Abstractions in a VLSI design flow.

The colored boxes on either side show the example Trojans in that level.

1) At the system level different hardware modules, interconnections and communication protocols used are defined. At this level, the Trojans may be triggered by the modules in the target hardware.

2) A typical development environment includes synthesis, simulation, verification and validation tools. The CAD tools and the scripts have been used to insert Trojans [10]. Software Trojans inserted in these CAD tools may mask the effects of the hardware Trojans.

3) At the RT level each functional module is described in terms of registers and signals. A trojan can be easily designed at the RT level as confirmed by the results to be discussed later.

4) At the gate level the design is represented as an interconnection of logic gates. This level allows an attacker to carefully control all aspects of the inserted trojan including size and location.

5) Transistors are used to build logic gates. This level gives the trojan designer control over circuit characteristics like power and timing. Individual transistors can be inserted or removed, altering the circuit functionality [11]. Transistor sizes can be modified to alter circuit parameters [11]. This is a very sophisticated attack, still in the trusted zone with difficult physical access.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

6) At the layout level, the dimensions and locations of all circuit components are described. This is the concrete level of the design where a trojan can be inserted. Trojans may be inserted by modifying the wire sizes, distances between circuit elements and re-assigning metal layers. Physical access is easier because of the untrusted zone. However, this hack has the highest level of sophistication.

HTH detection is an extremely challenging problem; traditional structural and functional tests do not seem to be effective in targeting and detecting HTHs. Since HTH can be introduced during different design phases, the nature of HTH differs from one design phase to the others. Therefore it is difficult to find a unique detection technique for all HTH. For instance Automatic Test Pattern Generation (ATPG) methods which are used in manufacturing test for detecting defects generally operate on the netlist of the HTH-free circuit. Existing ATPG algorithms cannot target HTH activation/detection directly [11] because HTH are designed such that they are silent most of their lifetime and have very small size relative to their host design, with featuring limited contribution into design characteristics. Such HTHs are most likely connected to nets with low controllability and/or observability[11],[12].

SIMULATION RESULTS

All the synthesis and simulation results are performed using Verilog HDL.Theproposed circuit has been simulated using Xilinx ISE14.4. The simulation results are shown below figures.Figures.

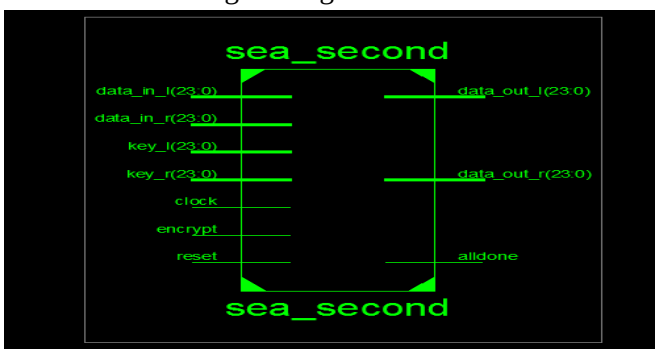


Fig.3: RTL schematic of Trojan in SEA crypto at Gate-Level

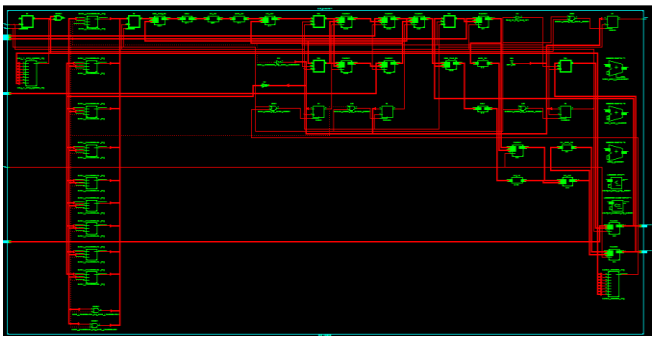
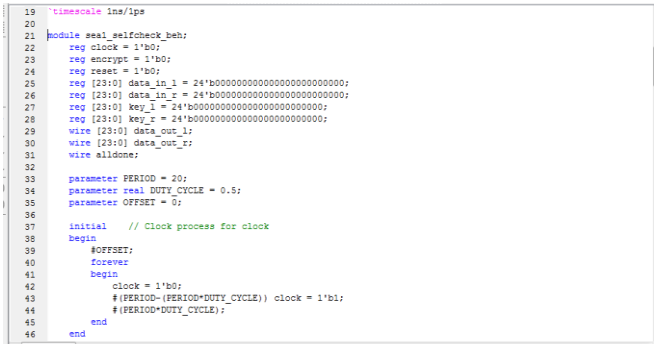


Fig.4: RTL sub schematic of Trojan in SEA crypto at Gate-Level



```
`timescale 1ns/1ps

module seal_selfcheck_beh;
    reg clock = 1'b0;
    reg encrypt = 1'b0;
    reg reset = 1'b0;
    reg [23:0] data_in_l = 24'b000000000000000000000000;
    reg [23:0] data_in_r = 24'b000000000000000000000000;
    reg [23:0] key_l = 24'b000000000000000000000000;
    reg [23:0] key_r = 24'b000000000000000000000000;
    wire [23:0] data_out_l;
    wire [23:0] data_out_r;
    wire alldone;

    parameter PERIOD = 20;
    parameter real DUTY_CYCLE = 0.5;
    parameter OFFSET = 0;

    initial    // Clock process for clock
    begin
        #OFFSET;
        forever
        begin
            clock = 1'b0;
            #(PERIOD-(PERIOD*DUTY_CYCLE)) clock = 1'b1;
            #(PERIOD*DUTY_CYCLE);
        end
    end
```

Fig.5: Test Bench of Trojan in SEA crypto at Gate-Level

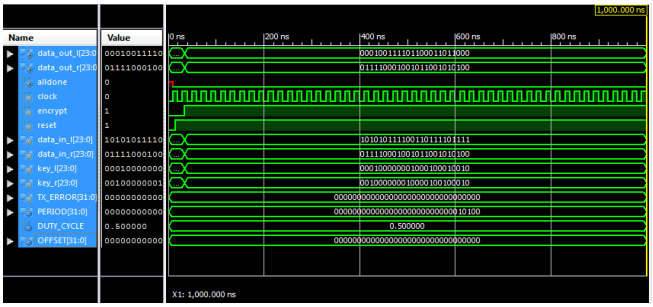


Fig.6: Simulation of Trojan in SEA crypto at Gate-Level

CONCLUSION AND FUTURE WORK

In this work we presented the possibility of insertion Trojan in SEA crypto at Gate-Level. Also we estimated the Trojan detection efficiency of path delay technique in architectures like SEA. By these results we proved that path delay technique is efficient to detection Trojan at Gate-Level. But, the increased delays are very small to detect at real time. This is the limitation of path delay technique. For our future work, first we would like to improve the Trojan

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

detection efficiency by talking the number of transitions at critical nodes along with the path delay. Second, we need to verify the possibility of Trojan insertion on SEA crypto at Register-Transfer Level (RTL).

REFERENCES

[1] F. Mace, F.-X.Standaert, and J.-J. Quisquaterr, "ASIC Implementations of the Block Cipher SEA for Constrained Applications". In Proc. of RFIDSEC'07, pp. 103-114, Malanga, Spain, 2007.

[2] F.-X. Standaert, G. Piret, N. Gershenfeld, J.-J.Quisquater, "SEA: A Scalable Encryption Algorithm for Small Embedded Applications". In Proc. of CARDIS 2006, LNCS ,pp 222- 236, Tarragona, Spain, April 2006.

[3] S. Bhasin, J-L.Danger, S. Guilley, XuanThuy Ngo and Laurent Sauvage, "Hardware Trojan Horses in Cryptographic IP Cores".In Proc.Of Fault Diagnosis and Tolerance in Cryptography (FDTC), 2013, pp. 15-29.

[4] Yier Jin and YiorgosMakris, "Hardware Trojan Detection Using Path Delay Fingerprint". In proc. Of Hardware-Oriented Security and Trust (HOST), 2008,June, pp. 51 – 57.

[5] Yier Jin and YiorgosMakris, "Hardware Trojan in Wireless Cryptographic ICs". In Proc. Of Design & Test of Computers, IEEE (Volume:27), Jan.-Feb. 2010, pp.26 – 35.

[6] DakshiAgrawal, SelcukBaktir, DenizKarakoyunlu, PankajRohatgi, and BerkSunar, "Trojan detection using ic fingerprinting," in Security and Privacy, 2007. SP '07. IEEE Symposium on, 2007, pp. 296–310.

[7] C. Bradford Barber, David P. Dobkin, and HannuHuhdanpaa, "The quickhull algorithm for convex hulls," ACM Trans. Math.Softw., vol. 22, no. 4, pp. 469–483, 1996.

[8] M. Tehranipoor and F. Koushanfar, ''A survey of hardware Trojan taxonomy and detection,'' IEEE Design Test Comput., pp. 10–25, Jan.–Feb. 2010.

[9] Rajendran et.al, "Trustworthy Hardware: Identifying and Classifying Hardware Trojans", In proc. of IEEE Computer Society,2010, vol.43, pp.39-46.

[10] J. A. Roy, F. Koushanfar, and I. L. Markov, "Extended abstract: Circuit cad tools as a security threat", In Proc. IEEE Workshop on Hardware Oriented Security and Trust, pages 65– 66, June 2008.

[11] Xiaoxiao Wang, M. Tehranipoor, and J. Plusquellic, "Detecting malicious inclusions in secure hardware: Challenges and solutions", In Hardware Oriented Security and Trust, 2008. HOST 2008. pp. 15–19, June 2008.

[12] MainakBanga and Michael S. Hsiao, "A Novel Sustained Vector Technique for the Detection of Hardware Trojans". In Proceedings of the 2009 22nd International Conference on VLSI Design, VLSID '09, pages 327–332, Washington, DC, USA, 2009. IEEE Computer Society.

[13] Paul C. Kocher, Joshua Jaffe and Benjamin Jun, "Differential Power Analysis". In Proceedings of CRYPTO'99, volume 1666 of LNCS, pages 388–397. Springer-Verlag, 1999.

[14] Jean-Jacques, Quisquater and David Samyde. "ElectroMagnetic Analysis (EMA): Measures and CounterMeasures for Smard Cards". In I. Attali and T. P. Jensen, editors, Smart Card Programming and Security (Esmart 2001), volume 2140 of LNCS, pages 200–210. Springer-Verlag, September 2001. Nice, France.

[15] S. Adee, ''The Hunt for the Kill Switch,'' IEEE Spectrum, vol. 45, no. 5, 2008, pp. 34-39. [16] TrustHub: http://trust-hub.org/